13. Fuzzy Rule-Based Systems

Luis Magdalena

Fuzzy rule-based systems are one of the most important areas of application of fuzzy sets and fuzzy logic. Constituting an extension of classical rule-based systems, these have been successfully applied to a wide range of problems in different domains for which uncertainty and vagueness emerge in multiple ways. In a broad sense, fuzzy rule-based systems are rule-based systems, where fuzzy sets and fuzzy logic are used as tools for representing different forms of knowledge about the problem at hand, as well as for modeling the interactions and relationships existing between its variables. The use of fuzzy statements as one of the main constituents of the rules allows capturing and handling the potential uncertainty of the represented knowledge. On the other hand, thanks to the use of fuzzy logic, inference methods have become more robust and flexible. This chapter will mainly analyze what is a fuzzy rule-based system (from both conceptual and structural points of view), how is it built, and how can be used. The analysis will start by considering the two main conceptual components of these systems, knowledge, and reasoning, and how they are represented. Then, a review of the main structural approaches to fuzzy rule-based systems will be considered. Hierarchical fuzzy systems will also be analyzed. Once defined the components, structure and approaches to those systems, the question of design will be considered. Finally, some conclusions will be presented.

From the point of view of applications, one of the most important areas of fuzzy sets theory is that of fuzzy rule-based systems (FRBSs). These kind of systems constitute an extension of classical rule-based systems, considering *IF–THEN* rules whose antecedents and consequents are composed of fuzzy logic (FL) statements, instead of classical logic ones.

Conventional approaches to knowledge representation are based on bivalent logic, which has associated a serious shortcoming: the inability to reason in situations of uncertainty and imprecision. As a consequence, conventional approaches do not provide an adequate framework for this mode of reasoning familiar to humans, and most commonsense reasoning falls into this category.

In a broad sense, an FRBS is a rule-based system where fuzzy sets and FL are used as tools for representing different forms of knowledge about the problem at hand, as well as for modeling the interactions and relationships existing between its variables.

The use of fuzzy statements as one of the main constituents of the rules, allows capturing and handling the potential uncertainty of the represented knowledge. On the other hand, thanks to the use of fuzzy logic, inference methods have become more robust and flexible.

Due to these properties, FRBSs have been successfully applied to a wide range of problems in different domains for which uncertainty and vagueness emerge in multiple ways [13.1–5].

The analysis of FRBSs will start by considering the two main conceptual components of these systems, knowledge and reasoning, and how they are presented. Then, a review of the main structural approaches to FRBSs will be considered. Hierarchical fuzzy systems would probably match in this previous section, but being possible to combine the hierarchical approach with any of the structural models defined there, it seems better to consider it independently. Once defined the components, structure, and approaches to those systems, the question of design will be considered. Finally, some conclusions will be presented. It is important to notice that this chapter will concentrate on the general aspects related to FRBSs without deepening in the foundations of FL which are widely considered in previous chapters.

13.1 Components of a Fuzzy Rule-Based System

Knowledge representation in FRBSs is enhanced with the use of linguistic variables and their linguistic values, that are defined by context-dependent fuzzy sets whose meanings are specified by gradual membership functions [13.6–8]. On the other hand, FL inference methods such as generalized Modus Ponens, generalized Modus Tollens, etc., form the basis for approximate reasoning [13.9]. Hence, FL provides a unique computational framework for inference in rule-based systems. This idea implies the presence of two clearly different concepts in FRBSs: knowledge and reasoning. This clear separation between knowledge and reasoning (the *knowledge base* (KB) and processing structure shown in Fig. 13.1) is the key aspect of knowledge-based systems, so that from this point of view, FRBSs can be considered as a type of knowledge-based system.

The first implementation of an FRBS dealing with real inputs and outputs was proposed by *Mamdani* [13.10], who considering the ideas published just a few months before by *Zadeh* [13.9] was able to augment his initial formulation allowing the application of fuzzy systems (FSs) to a control problem, so creating the first fuzzy control application. These kinds of FSs are also referred to as FRBSs with fuzzifier and defuzzifier or, more commonly, as *fuzzy logic controllers* (FLCs), as proposed by the author in his pioneering paper [13.11], or Mamdani FRBSs. From the beginning, the term FLC became popular since control systems design constituted the main application of Mamdani FRBSs. At present, control is only one more of the many application areas of FRBSs.

The generic structure of a Mamdani FRBS is shown in Fig. 13.1. The KB stores the available knowledge about the problem in the form of fuzzy *IF–THEN* rules. The processing structure, by means of these rules, puts into effect the inference process on the system inputs.

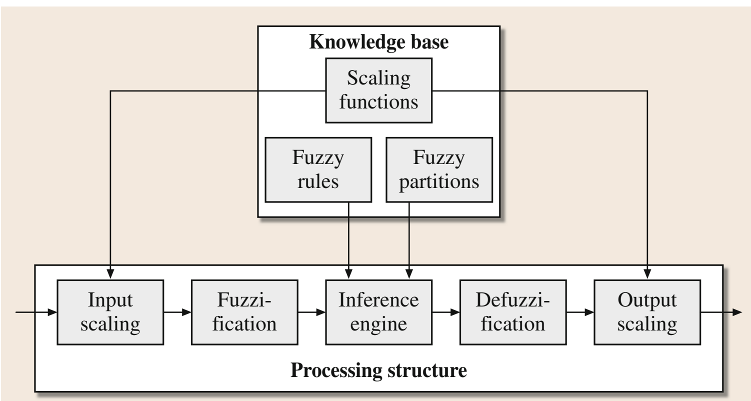

Fig. 13.1 General structure of a Mamdani FRBS

The fulfillment of rule antecedent gives rise to the execution of its consequent, i. e., one output is produced. The overall process includes several steps. The input and output scalings produce domain adaptations. Fuzzification interface establishes a mapping between crisp values in the input domain U, and fuzzy sets defined on the same universe of discourse. On the other hand, the defuzzification interface performs the opposite operation by defining a mapping between fuzzy sets defined in the output domain V and crisp values defined in the same universe. The central step of the process is inference.

The next two subsections analyze in depth the two main components of an FRBS, the KB and the processing structure, considering the case of a Mamdani FRBS.

13.1.1 Knowledge Base

The KB of an FRBS serves as the repository of the problem-specific knowledge – that models the relationship between input and output of the underlying system – upon which the inference process reasons to obtain from an observed input, an associated output.

This knowledge is represented in the form of rules, and the most common rule structure in Mamdani FRBSs involves the use of linguistic variables [13.6–8]. Hence, when dealing with multiple inputs-single output (MISO) systems, these *linguistic* rules possess the following form

$$\begin{aligned}&\mathit{IF}\ X_1\ \mathit{is}\ LT_1\ \mathit{and}\ \ldots\ \mathit{and}\ X_n\ \mathit{is}\ LT_n\\&\mathit{THEN}\ Y\ \mathit{is}\ LT_o\ ,\end{aligned} \tag{13.1}$$

with X_i and Y being, respectively, the input and output linguistic variables, and with LT_i being linguistic terms associated with these variables.

Note that the KB contains two different information levels, i. e., the linguistic variables (providing fuzzy rule semantics in the form of fuzzy partitions) and the linguistic rules representing the expert knowledge. Apart from that, a third component, scaling functions, is added in many FRBSs to act as an interfacing component for domain adaptation between the external world and the universes of discourse used at the level of the fuzzy partitions. This conceptual distinction drives to the three separate entities that constitute the KB:

- The *fuzzy partitions* (also called Frames of Cognition) describe the sets of linguistic terms associated with each variable and considered in the linguistic rules, and the membership functions defining the semantics of these linguistic terms. Each linguistic variable involved in the problem will have associated a fuzzy partition of its domain. Figure 13.2 shows a fuzzy partition using triangular membership functions. This structure provides a natural framework to include expert knowledge in the form of fuzzy rules. The fuzzy partition shown in the figure uses five linguistic terms {*very small, small, medium, large*, and *very large*}, (represented as VS, S, M, L, and VL, respectively) with the interval $[l, r]$ being its domain (Universe of discourse). The figure also shows the membership function associated to each of these five terms.
- A *rule base* (RB) is comprised of a collection of linguistic rules (as the one shown in (13.1)) that are joined by the *also* operator. In other words, multiple rules can fire simultaneously for the same input.
- Moreover, the KB also comprises the *scaling functions* or scaling factors that are used to transform between the universe of discourse in which the fuzzy sets are defined from/to the domain of the system input and output variables.

It is important to note that the RB can present several structures. The usual one is the list of rules, although a decision table (also called rule matrix) becomes an equivalent and more compact representation for the same set of linguistic rules when only a few input variables (usually one or two) are considered by the FRBS.

Let us consider an FRBS where two input variables (x_1 and x_2) and a single output variable (y) are involved, with the following term sets associated: {*small, medium, large*}, {*short, medium, long*} and {*bad, medium, good*}, respectively. The following RB

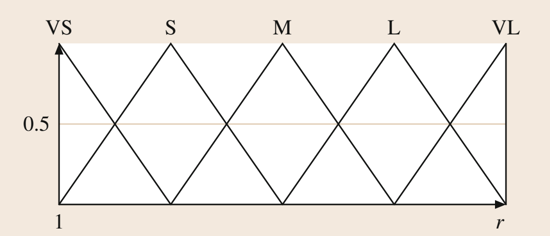

Fig. 13.2 Example of a fuzzy partition

Part B | 13.1

composed of five linguistic rules

R_1: *IF* X_1 *is small and* X_2 *is short THEN Y is bad, also*
R_2: *IF* X_1 *is small and* X_2 *is medium THEN Y is bad, also*
R_3: *IF* X_1 *is medium and* X_2 *is short THEN Y is medium, also*
R_4: *IF* X_1 *is large and* X_2 *is medium THEN Y is medium, also*
R_5: *IF* X_1 *is large and* X_2 *is long THEN Y is good* , (13.2)

can be represented by the decision table shown in Table 13.1.

Before concluding this section, we should notice two aspects. On one hand, the structure of a linguistic rule may be more generic if a connective other than the *and* operator is used to aggregate the terms in the rule antecedent. However, it has been demonstrated that the above rule structure is generic enough to subsume other possible rule representations [13.12]. The above rules are therefore commonly used throughout the literature due to their simplicity and generality. On the other hand, linguistic rules are not the only option and rules with a different structure can be considered, as we shall see in Sect. 13.2.

13.1.2 Processing Structure

The functioning of FRBSs has been described as the interaction of knowledge and reasoning. Once briefly considered the knowledge component, this section will analyze the reasoning (processing) structure. The processing structure of a Mamdani FRBS is composed of the following five components:

- The *input scaling* that transforms the values of the input variables from its domain to the one where the input fuzzy partitions are defined.
- A *fuzzification interface* that transforms the crisp input data into fuzzy values that serve as the input to the fuzzy reasoning process.
- An *inference engine* that infers from the fuzzy inputs to several resulting output fuzzy sets according to the information stored in the KB.
- A *defuzzification interface* that converts the fuzzy sets obtained from the inference process into a crisp value.
- The *output scaling* that transforms the defuzzified value from the domain of the output fuzzy partitions to that of the output variables, constituting the global output of the FRBS.

Table 13.1 Example of a decision table

	x_1		
x_2	***small***	***medium***	***large***
short	bad	medium	
medium	bad		medium
long			good

In the following, the five elements will be briefly described.

The Input/Output Scaling

Input/output scaling maps (applying the corresponding scaling functions or factors contained in the KB) the input/output variables to/from the universes of discourse over which the corresponding linguistic variables were defined.

This mapping can be performed with different functions ranging from a simple scaling factor to linear and nonlinear functions.

The initial idea for scaling was the use of *scaling factors* with a tuning purpose [13.13], giving a certain adaptation capability to the fuzzy system.

Additional degrees of freedom could be obtained by using a more complex scaling function. A second option is the use of *linear scaling* with a function of the form

$$f(x) = \lambda \cdot x + \nu \,, \tag{13.3}$$

where the scaling factor λ enlarges or reduces the operating range, which in turn decreases or increases the sensitivity of the system in respect to that input variable, or the corresponding gain in the case of an output variable. The parameter ν shifts the operating range and plays the role of an offset for the corresponding variable.

Finally, it is possible to use more complex mappings generating *nonlinear scaling*. A common nonlinear scaling function is

$$f(x) = \text{sign}(x) \cdot |x|^{\alpha} \,. \tag{13.4}$$

This nonlinear scaling increases ($\alpha > 1$) or decreases ($\alpha < 1$) the relative sensitivity in the region closer to the central point of the interval and has the opposite effect when moving far from the central point [13.14].

The Fuzzification Interface

The fuzzification interface enables Mamdani FRBSs to handle crisp input values. Fuzzification establishes a mapping from crisp input values to fuzzy sets defined in the universe of discourse of those inputs. The membership function of the fuzzy set A' defined over the universe of discourse U associated to a crisp input value x_0 is computed as

$$\mu_{A'} = F(x_0) , \tag{13.5}$$

in which F is a fuzzification operator.

The most common choice for the fuzzification operator F is the *point wise* or *singleton fuzzification*, where A' is built as a singleton with support x_0, i. e., it presents the following membership function:

$$\mu_{A'}(x) = \begin{cases} 1, & \text{if } x = x_0 \\ 0, & \text{otherwise} . \end{cases} \tag{13.6}$$

Nonsingleton options [13.15] are also possible and have been considered in some cases as a tool to represent the imprecision of measurements.

The Inference System

The inference system is the component that derives the fuzzy outputs from the input fuzzy sets according to the relation defined through the fuzzy rules. The usual fuzzy inference scheme employs the generalized Modus Ponens, an extension to the classical Modus Ponens [13.9]

$$\begin{array}{l} \textit{IF X is A THEN Y is B} \\ \textit{X is } A' \\ \hline \qquad\qquad \textit{Y is } B' . \end{array} \tag{13.7}$$

In this expression, *IF X is A THEN Y is B* describes a conditional statement that in this case is a fuzzy conditional statement, since A and B are fuzzy sets, and X and Y are linguistic variables. A fuzzy conditional statement like this one represents a fuzzy relation between A and B defined in $U \times V$. This fuzzy relation is expressed again by a fuzzy set (R) whose membership function $\mu_R(x, y)$ is given by

$$\mu_R(x, y) = I(\mu_A(x), \mu_B(y)), \quad \forall x \in U, y \in V , \tag{13.8}$$

in which $\mu_A(x)$ and $\mu_B(y)$ are the membership functions of the fuzzy sets A and B, and I is a fuzzy implication operator that models the existing fuzzy relation.

Going back to (13.7), the result of applying generalized Modus Ponens is obtaining the fuzzy set B' (through its membership function) by means of the *compositional rule of inference* [13.9]:

> *If R is a fuzzy relation defined in U and V, and A′ is a fuzzy set defined in U, then the fuzzy set B′, induced by A′, is obtained from the composition of R and A′,*

that is

$$B' = A' \circ R . \tag{13.9}$$

Now it is needed to compute the fuzzy set B' from A' and R. According to the definition of composition (*T-composition*) given in the chapter devoted to fuzzy relations, the result will be

$$\mu_{B'}(y) = \sup_{x \in U} T(\mu_{A'}(x), \mu_R(x, y)) , \tag{13.10}$$

where T is a triangular norm (t-norm). The concept and properties of t-norms have been previously introduced in the chapter devoted to fuzzy sets.

Given now an input value $X = x_0$, obtaining A' in accordance with (13.6) (where $\mu_{A'}(x) = 0 \ \ \forall x \neq x_0$), and considering the properties of t-norms ($T(1, a) = a, T(0, a) = 0$), the previous expression is reduced to

$$\begin{aligned} \mu_{B'}(y) &= T(\mu_{A'}(x_0), \mu_R(x_0, y)) \\ &= T(1, \mu_R(x_0, y)) = \mu_R(x_0, y) . \end{aligned} \tag{13.11}$$

The only additional point to arrive to the final value of $\mu_{B'}(y)$ is the definition of R, the fuzzy relation representing the Implication. This is a somehow controversial question. Since the very first applications of FRBSs [13.10, 11] this relation has been implemented with the minimum (product has been also a common choice). If we analyze the definition of fuzzy implication given in the corresponding chapter, it is clear that the minimum does not satisfy all the conditions to be a fuzzy implication, so, why is it used? It can be said that initially it was a short of heuristic decision, which demonstrated really good results being accepted and reproduced in all subsequent applications. Further analysis can offer different explanations to this choice [13.16–18].

In any case, assuming the minimum as the representation for R, (13.11) produces the following final result:

$$\mu_{B'}(y) = \min(\mu_A(x_0), \mu_B(y)) . \tag{13.12}$$

Considering now an n-dimensional input space, the inference will establish a mapping between fuzzy sets defined in the Cartesian product ($U = U_1 \times U_2 \times \cdots \times U_n$) of the universes of discourse of the input variables $X_1, \ldots, X_n$, and fuzzy sets defined in V, being the universe of discourse of the output variable Y. Therefore, when applied to the ith rule of the RB, defined as

$$R_i : \textit{IF } X_1 \textit{ is } A_{i1} \textit{ and } \ldots \textit{ and } X_n \textit{ is } A_{in} \textit{ THEN } Y \textit{ is } B_i\ , \tag{13.13}$$

considering an input value $x_0 = (x_1, \ldots, x_n)$, the output fuzzy set B' will be obtained by replacing $\mu_A(x_0)$ in (13.12) with,

$$\mu_{A_i}(x_0) = T(\mu_{A_{i1}}(x_1), \ldots, \mu_{A_{in}}(x_n))\ ,$$

where T is a fuzzy conjunctive operator (a t-norm).

The Defuzzification Interface

The inference process in Mamdani-type FRBSs operates at the level of individual rules. Thus, the application of the compositional rule of inference to the current input, using the m rules in the KB, generates m output fuzzy sets B'_i. The defuzzification interface has to aggregate the information provided by the m individual outputs and obtain a crisp output value from the aggregated set. This task can be done in two different ways [13.1, 12, 19]: *Mode A-FATI* (first aggregate, then infer) and *Mode B-FITA* (first infer, then aggregate).

Mamdani originally suggested the mode A-FATI in his first conception of FLCs [13.10]. In the last few years, the Mode B-FITA is becoming more popular [13.19–21], in particular, in real-time applications which demand a fast response time.

Mode A-FATI: First Aggregate, then Infer. In this case, the defuzzification interface operates as follows:

- Aggregate the individual fuzzy sets B'_i into an overall fuzzy set B' by means of a fuzzy aggregation operator G (usually named as the *also* operator):

$$\mu_{B'}(y) = G\left\{\mu_{B'_1}(y), \mu_{B'_2}(y), \ldots, \mu_{B'_m}(y)\right\}\ . \tag{13.14}$$

- Employ a defuzzification method, D, transforming the fuzzy set B' into a crisp output value y_0:

$$y_0 = D(\mu_{B'}(y))\ . \tag{13.15}$$

Usually, the aggregation operator G is implemented by the maximum (a t-conorm), and the defuzzifier D is the *center of gravity* (CG) or the *mean of maxima* (MOM), whose expressions are as follows:

- CG:

$$y_0 = \frac{\int_Y y \cdot \mu_{B'}(y)\mathrm{d}y}{\int_Y \mu_{B'}(y)\mathrm{d}y}\ . \tag{13.16}$$

- MOM:

$$y_{\inf} = \inf\{z | \mu_{B'}(z) = \max_y \mu_{B'}(y)\}$$
$$y_{\sup} = \sup\{z | \mu_{B'}(z) = \max_y \mu_{B'}(y)\}$$
$$y_0 = \frac{y_{\inf} + y_{\sup}}{2}\ . \tag{13.17}$$

Mode B-FITA: First Infer, then Aggregate. In this second approach, the contribution of each fuzzy set is considered separately and the final crisp value is obtained by means of an averaging or selection operation applied to the set of crisp values derived from each of the individual fuzzy sets B'_i.

The most common choice is either the CG or the *maximum value* (MV), then weighted by the matching degree. Its expression is shown as follows:

$$y_0 = \frac{\sum_{i=1}^m h_i \cdot y_i}{\sum_{i=1}^m h_i}\ , \tag{13.18}$$

with y_i being the CG or the MV of the fuzzy set B'_i, inferred from rule R_i, and $h_i = \mu_{A_i}(x_0)$ being the matching between the system input x_0 and the antecedent (premise) of rule i.

Hence, this approach avoids aggregating the rule outputs to generate the final fuzzy set B', reducing the computational burden compared to mode A-FATI defuzzification.

This defuzzification mode constitutes a different approach to the notion of the *also* operator, and it is directly related to the idea of interpolation and the approach of Takagi–Sugeno–Kang (TSK) fuzzy systems, as can be seen by comparing (13.18) and (13.25).

13.2 Types of Fuzzy Rule-Based Systems

As discussed earlier, the first proposal of an FRBS was that of Mamdani, and this kind of system has been considered as the basis for the general description of previous section. This section will focus on the different structures that can be considered when building an FRBS.

13.2.1 Linguistic Fuzzy Rule-Based Systems

This approach corresponds to the original *Mamdani* FRBS [13.10, 11], being the main tool to develop Linguistic models, and is the approach that has been mainly considered to this point in the chapter.

A Mamdani FRBS provides a natural framework to include expert knowledge in the form of linguistic rules. This knowledge can be easily combined with rules which are automatically generated from data sets that describe the relation between system input and output. In addition, this knowledge is highly interpretable. The fuzzy rules are composed of input and output variables, which take values from their term sets having a meaning (a semantics) associated with each linguistic term. Therefore, each rule is a description of a condition-action statement that exhibits a clear interpretation to a human – for this reason, these kinds of systems are usually called *linguistic* or *descriptive Mamdani FRBSs*. This property makes Mamdani FRBSs appropriate for applications in which the emphasis lies on model interpretability, such as fuzzy control [13.20, 22, 23] and linguistic modeling [13.4, 21].

13.2.2 Variants of Mamdani Fuzzy Rule-Based Systems

Although Mamdani FRBSs possess several advantages, they also come with some drawbacks. One of the problems, especially in linguistic modeling applications, is their limited accuracy in some complex problems, which is due to the structure of the linguistic rules. [13.24] and [13.25] analyzed these limitations concluding that the structure of the fuzzy linguistic *IF–THEN* rule is subject to certain restrictions because of the use of linguistic variables:

- There is a lack of flexibility in the FRBS due to the rigid partitioning of the input and output spaces.
- When the input variables are mutually dependent, it becomes difficult to find a proper fuzzy partition of the input space.
- The homogeneous partition of the input and output space becomes inefficient and does not scale well as the dimensionality and complexity of the input–output mapping increases.
- The size of the KB increases rapidly with the number of variables and linguistic terms in the system. This problem is known as the *course of dimensionality*. In order to obtain an accurate FRBS, a fine level of granularity is needed, which requires additional linguistic terms. This increase in granularity causes the number of rules to grow, which complicates the interpretability of the system by a human. Moreover, in the vast majority of cases, it is possible to obtain an equivalent FRBS that achieves the same accuracy with a fewer number of rules whose fuzzy sets are not restricted to a fixed input space partition.

Both variants of linguistic Mamdani FRBSs described in this section attempt to solve the said problems by making the linguistic rule structure more flexible.

DNF Mamdani Fuzzy Rule-Based Systems

The first extension to Mamdani FRBSs aims at a different rule structure, the so-called *disjunctive normal form (DNF) fuzzy rule*, which has the following form [13.26, 27]:

$$\begin{aligned}&IF\ X_1\ is\ \widetilde{A_1}\ and\ \ldots\ and\ X_n\ is\ \widetilde{A_n}\\&THEN\ Y\ is\ B\,,\end{aligned} \tag{13.19}$$

where each input variable X_i takes as its value a set of linguistic terms $\widetilde{A_i}$, whose members are joined by a disjunctive operator, while the output variable remains a usual linguistic variable with a single label associated. Thus, the complete syntax for the antecedent of the rule is

$$\begin{aligned}&X_1\ is\ \widetilde{A_1} = \{A_{11}\ or\ \ldots\ or\ A_{1l_1}\}\ and\ \ldots\\&and\ X_n\ is\ \widetilde{A_n} = \{A_{n1}\ or\ \ldots\ or\ A_{nl_n}\}\,.\end{aligned} \tag{13.20}$$

An example of this kind of rule is shown as follows. Let us suppose we have three input variables, X_1, X_2, and X_3, and one output variable, Y, such that the linguistic term sets D_i $(i = 1, 2, 3)$ and F, associated with each variable, are

$$D_1 = \{A_{11}, A_{12}, A_{13}\}$$
$$D_2 = \{A_{21}, A_{22}, A_{23}, A_{24}, A_{25}\}$$
$$D_3 = \{A_{31}, A_{32}\} \quad F = \{B_1, B_2, B_3\} . \tag{13.21}$$

In this case, an example of DNF rule will be

$$\textit{IF } X_1 \textit{ is } \{A_{11} \textit{ or } A_{12}\} \textit{ and } X_2 \textit{ is } \{A_{23} \textit{ or } A_{24}\}$$
$$\textit{and } X_3 \textit{ is } \{A_{31} \textit{ or } A_{32}\} \textit{ THEN } Y \textit{ is } B_2 . \tag{13.22}$$

This expression contains an additional *connective* different than the *and* considered in all previous rules. The *or* connective is computed through a t-conorm, the maximum being the most commonly used.

The main advantage of this rule structure is its ability to integrate in a single expression (a single DNF rule) the information corresponding to several elemental rules (the rules commonly used in Mamdani FRBSs). In this example, (13.22) corresponds to 8 $(2 \times 2 \times 2)$ rules of the equivalent system expressed as (13.1). This property produces a certain level of compression of the rule base, being quite helpful when the number of input variables increases, alleviating the effect of the course of dimensionality.

Approximate Mamdani-Type Fuzzy Rule-Based Systems

While the previous DNF fuzzy rule structure does not involve an important loss in the linguistic Mamdani FRBS interpretability, the point of departure for the second extension is to obtain an FRBS which achieves a better accuracy at the cost of reduced interpretability. These systems are called *approximate Mamdani-type FRBSs* [13.1, 25, 28–30], in opposition to the previous *descriptive* or *linguistic Mamdani FRBSs*.

The structure of an approximate FRBS is similar to that of a descriptive one shown in Fig. 13.1. The difference is that in this case, the rules do not refer in their definition to predefined fuzzy partitions of the linguistic variables. In an approximate FRBS, each rule defines its own fuzzy sets instead of using a linguistic label pointing to a particular fuzzy set of the partition of the underlying linguistic variable. Thus, an approximate fuzzy rule has the following form:

$$\textit{IF } X_1 \textit{ is } A_1 \textit{ and } \ldots \textit{ and } X_n \textit{ is } A_n \textit{ THEN } Y \textit{ is } B . \tag{13.23}$$

The major difference with respect to the rule structure considered in linguistic Mamdani FRBSs is the fact that the input variables X_i and the output one Y are fuzzy variables instead of linguistic variables and, thus, A_i and B are not linguistic terms (LT_i) as they were in (13.1), but independently defined fuzzy sets that elude an intuitive linguistic interpretation. In other words, rules of approximate nature are *semantic free*, whereas descriptive rules operate in the context formulated by means of the linguistic terms semantics.

Therefore, approximate FRBSs do not relay on fuzzy partitions defining a semantic context in the form of linguistic terms. The fuzzy partitions are somehow integrated into the fuzzy rule base in which each rule subsumes the definition of its underlying input and output fuzzy sets, as shown in Fig. 13.3(b).

Approximate FRBSs demonstrate some specific advantages over linguistic FRBSs making them particularly useful for certain types of applications [13.25]:

- The major advantage of the approximate approach is that each rule employs its own distinct fuzzy sets resulting in additional degrees of freedom and an increase in expressiveness. It means that the tuning of a certain fuzzy set in a rule will have no effect on other rules, while changing a fuzzy set of a fuzzy partition in a descriptive model affects all rules considering the corresponding linguistic label.
- Another important advantage is that the number of rules can be adapted to the complexity of the problem. Simple input–output relationships are modeled with a few rules, but still more rules can be added as

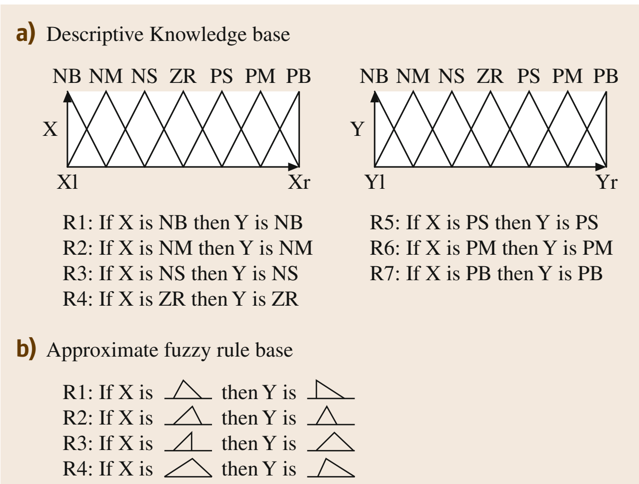

Fig. 13.3a,b Comparison between a descriptive KB and an approximate fuzzy rule base

the complexity of the problem increases. Therefore, approximate FRBSs constitute a potential remedy to the course of dimensionality that emerges when scaling to multidimensional systems.

These properties enable approximate FRBSs to achieve a better accuracy than linguistic FRBS in complex problem domains. However, despite their benefits, they also come with some drawbacks:

- Their main drawback compared to the descriptive FRBS is the degradation in terms of interpretability of the RB as the fuzzy variables no longer share a unique linguistic interpretation. Still, unlike other kinds of approximate models such as neural networks that store knowledge implicitly, the knowledge in an approximate FRBS remains explicit as the system behavior is described by local rules. Therefore, approximate FRBSs can be considered as a compromise between the apparent interpretability of descriptive FRBSs and the type of black-box behavior, typical for nondescriptive, implicit models.
- The capability to approximate a set of training data accurately can lead to over-fitting and therefore to a poor generalization capability to cope with previously unseen input data.

According to their properties, fuzzy modeling [13.1] constitutes the major application of approximate FRBSs, as model accuracy is more relevant than description ability. Approximate FRBSs are usually not the first choice for linguistic modeling and fuzzy control problems. Hence, descriptive and approximate FRBSs are considered as complementary rather than competitive approaches. Depending on the problem domain and requirements on the obtained model, one should use one or the other approach. Approximate FRBSs are recommendable in case one wants to trade interpretability for improved accuracy.

13.2.3 Takagi–Sugeno–Kang Fuzzy Rule-Based Systems

Instead of working with linguistic rules of the kind introduced in the previous section, *Sugeno* et al. [13.31, 32] proposed a new model based on rules whose antecedent is composed of linguistic variables and the consequent is represented by a function of the input variables. The most common form of this kind of rules is the one in which the consequent expression constitutes a linear combination of the variables involved in the antecedent

$$\begin{aligned}&\mathit{IF}\ X_1\ \mathit{is}\ A_1\ \mathit{and}\ \dots\ \mathit{and}\ X_n\ \mathit{is}\ A_n\\&\mathit{THEN}\ Y=p_0+p_1\cdot X_1+\cdots+p_n\cdot X_n\,,\end{aligned}\qquad(13.24)$$

where X_i are the input variables, Y is the output variable, and $p=(p_0,p_1,\dots,p_n)$ is a vector of real parameters. Regarding A_i, they are either a direct specification of a fuzzy set (thus X_i being fuzzy variables) or a linguistic label that points to a particular member of a fuzzy partition of a linguistic variable. These rules, and consequently the systems using them, are usually called *TSK fuzzy rules*, in reference to the names of their first proponents.

The output of a TSK FRBS, using a KB composed of m rules, is obtained as a weighted sum of the individual outputs provided by each rule, Y_i, $i=1,\dots,m$, as follows:

$$\frac{\sum_{i=1}^{m} h_i\cdot Y_i}{\sum_{i=1}^{m} h_i}\,,\qquad(13.25)$$

in which $h_i=T(A_{i1}(x_1),\dots,A_{in}(x_n))$ is the matching degree between the antecedent part of the ith rule and the current inputs to the system, $x_0=(x_1,\dots,x_n)$. T stands for a conjunctive operator modeled by a t-norm. Therefore, to design the inference engine of TSK FRBSs, the designer only selects this conjunctive operator T, with the most common choices being the minimum and the product. As a consequence, TSK systems do not need defuzzification, being their outputs real numbers.

This type of FRBS divides the input space in several fuzzy subspaces and defines a linear input–output relationship in each one of these subspaces [13.31]. In the inference process, these partial relationships are combined in the said way for obtaining the global input–output relationship, taking into account the dominance of the partial relationships in their respective areas of application and the conflicts emerging in the overlapping zones. As a result, the overall system performs as a sort of interpolation of the local models represented by each individual rule.

TSK FRBSs have been successfully applied to a large variety of practical problems. The main advantage of these systems is that they present a set of compact system equations that allows the parameters p_i to be estimated by means of classical regression methods, which facilitates the design process. However, the main drawback associated with TSK FRBSs is the form of the rule consequents, which does not provide a natural framework for representing expert knowledge that is afflicted with uncertainty. Still, it becomes possible to integrate expert knowledge in these FRBSs by slightly modifying the rule consequent: for each linguistic rule with consequent *Y is B*, provided by an expert, its consequent is substituted by $Y = p_0$, with p_0 standing for the modal point of the fuzzy set associated with the label B. These kinds of rules are usually called *simplified TSK rules* or *zero-order TSK rules*.

However, TSK FRBSs are more difficult to interpret than Mamdani FRBSs due to two different reasons:

- The structure of the rule consequents is difficult to be understood by human experts, except for zero-order TSK.
- Their overall output simultaneously depends on the activation of the rule antecedents and on the evaluation of the function defining rule consequent, that depends itself on the crisp inputs as well, rather than being constant.

TSK FRBSs are used in fuzzy modeling [13.4, 31] as well as control problems [13.31, 33].

As with Mamdani FRBSs, it is also possible to built descriptive as well as approximate TSK systems.

13.2.4 Singleton Fuzzy Rule-Based Systems

The singleton FRBS, where the rule consequent takes a single real-valued number, may be considered as a particular case of the linguistic FRBS (the consequent is a fuzzy set where the membership function is one for a specific value and zero for the remaining ones) or of the TSK-type FRBS (the previously described zero-order TSK systems).

Its rule structure is the following

$$\begin{aligned}&\textit{IF } X_1 \textit{ is } A_1 \textit{ and } \ldots \textit{ and } X_n \textit{ is } A_n\\&\textit{THEN } Y \textit{ is } y_0 \,.\end{aligned} \tag{13.26}$$

Since the single consequent seems to be more easily interpretable than a polynomial function, the singleton FRBS may be used to develop linguistic fuzzy models. Nevertheless, compared with the linguistic FRBS, the fact of having a different consequent value for each rule (no global semantic is used for the output variable) worsens the interpretability.

13.2.5 Fuzzy Rule-Based Classifiers

Previous sections have implicitly considered FRBSs working with inputs and, what is more important, outputs which are real variables. These kinds of fuzzy systems show an interpolative behavior where the overall output is a combination of the individual outputs of the fired rules. This interpolative behavior is explicit in TSK models but it is also present in Mamdani systems. This situation gives FRBSs a sort of smooth output, generating soft transitions between rules, and being one of the significant properties of FRBSs.

A completely different situation is that of having a problem where the output takes values from a finite list of possible values representing categories or classes. Under those circumstances, the interpolative approach of previously defined aggregation and defuzzification methods, makes no sense. As a consequence, some additional comments will be added to highlight the main characteristics of fuzzy rule-based classifiers (FRBCs), and the differences with other FRBSs.

A fuzzy rule-based classifier is an automatic classification system that uses fuzzy rules as knowledge representation tool. Therefore, the fuzzy classification rule structure is as follows

$$\begin{aligned}&\textit{IF } X_1 \textit{ is } A_1 \textit{ and } \ldots \textit{ and } X_n \textit{ is } A_n\\&\textit{THEN } Y \textit{ is } C \,,\end{aligned} \tag{13.27}$$

with Y being a categorical variable, so C being a class label. The processing structure is similar to that previously described in what concerns to the evaluation of matching degree between each rule's antecedent and current input, i. e., for each rule R_i we obtain $h_i = T(A_{i1}(x_1), \ldots, A_{in}(x_n))$. Once obtained h_i, the winner rule criteria could be applied so that the overall output is assigned with the consequent of the rule achieving the highest matching degree (highest value of h_i). More elaborated evaluations as voting are also possible.

Other alternative representations that include a certainty degree or weight for each rule have also been considered [13.34]. In this case, the previously described rule will also include a rule weight w_i that weights the matching degree during the inference process. The effect will be that the winning rule will be that achieving the highest value of $h_i \cdot w_i$, or in the case of voting schemes, the influence of the vote of the rule will be proportional to this value.

13.2.6 Type-2 Fuzzy Rule-Based Systems

The idea of extending fuzzy sets by allowing membership functions to include some kind of uncertainty was already mentioned by *Zadeh* in early papers [13.6–8]. The idea, that was not really exploited for a long period, has achieved now a significant presence in the literature with the proposal of Type-2 fuzzy systems and Interval type-2 fuzzy systems [13.35]. The main concept is that the membership degree is not a value but a fuzzy set or an interval, respectively. The effect is obtaining additional degrees of freedom being available in the design process, but increasing the complexity of the processing structure that requires now a type-reduction step added to the overall process described in previous section. As the complexity of the type reduction process is much lower for Interval type-2 fuzzy systems than in the general case of type-2 fuzzy systems, interval approaches are the most widely considered now in the area of Type-2 fuzzy sets.

13.2.7 Fuzzy Systems with Implicative Rules

Rule-based systems mentioned to this point consider rules that, having the form *if X is A then Y is B*, model the inference through a t-norm, usually minimum or product (Sect. 13.1.2). With this interpretation, rules are described as conjunctive rules, representing joint sets of possible input and output values. As mentioned in the chapter devoted to fuzzy control, these rules should be seen not as logical implications but rather as input–output associations.

That kind of rule is the one commonly used in real applications to the date. However, different authors have pointed out that the same rule will have a completely different meaning when modeled in terms of material implications (the approach for Boolean *if–then* statements in propositional logic) [13.18]. As a result, in addition to the common interpretation of fuzzy rules that is widely considered in the literature, some authors are exploring the modeling of fuzzy rules (with exactly the same structure previously mentioned) by means of *material implications* [13.36]. Even being in a quite preliminary stage of development, it is interesting to mention this ideas since it constitutes a completely different interpretation of FRBSs, offering so new possibilities to the field.

13.3 Hierarchical Fuzzy Rule-Based Systems

The knowledge structure of FRBSs offers different options to introduce hierarchical structures. Rules, partitions, or variables can be distributed at different levels according to their specificity, granularity, relevance, etc. This section will introduce different approaches to hierarchical FRBSs.

It would be possible to consider hierarchical fuzzy systems as a different type of FRBS, so including it in previous section, or as a design option to build *simpler* FRBSs, being then included as part of the next section. Including it in previous section could be a little bit confusing since it is possible to combine the hierarchical approach with several of the structural models defined there, it seems better to consider it independently devoting a section to analyze them.

The definition of hierarchical fuzzy systems as a method to solve problems with a higher level of complexity than those usually focused on with FRBSs, has produced some good results. In most of the cases, the underlying idea is to cope with the complexity of a problem by applying some kind of decomposition that generates a hierarchy of lower complexity systems [13.37].

Several methods to establish hierarchies in fuzzy controllers have been proposed. These methods could be grouped according to the way they structure the inference process, and the knowledge applied.

A first approach defines the hierarchy as a prioritization of rules in such a way that rules with a different level of specificity receive a different priority, having higher priority those rules being more specific [13.38, 39]. With this kind of hierarchy, a generic rule is applied only when no suitable specific rule is available. In this case, the hierarchy is the effect of a particular implication mechanism applying the rules by taking into account its priority. This methodology defines the hierarchy (the decomposition) at the level of rules. The rules are grouped into prioritized levels to design a hierarchical fuzzy controller.

Another option is that of considering a hierarchy of fuzzy partitions with different granularity [13.40]. From that point, an FRBS is structured in layers, where each layer contains fuzzy partitions with a different granularity, as well as the rules using those fuzzy partitions. Usually, every partition in a certain layer has the same number of fuzzy terms. In this case, rules at different layers have different granularity, being somehow related to the idea of specificity of the previous paragraph. It is even possible to generate a multilevel grid-like partition where only for some specific regions of the input space (usually those regions showing poor performance) a higher granularity is considered [13.41], with a similar approach to that already considered in some neuro-fuzzy systems [13.42].

A completely different point of view is that of introducing the decomposition at the level of variables. In this case, the input space is decomposed into subspaces of lower dimensionality, and each input variable is only considered at a certain level of the hierarchy. The result is a cascade structure of FRBSs where, in addition to a subset of the input variables, the output of each level is considered as one of the inputs to the following level [13.43]. As a result, the system is decomposed into a finite number of reduced-order subsystems, eliminating the need for a large-sized inference engine. This decomposition is usually stated as a way to maintain under control the problems generated by the so-called course of dimensionality, the exponential growth of the number of rules related to the number of variables of the system.

The number of rules of an FRBS with n input variables and l linguistic terms per variable, will be l^n. In this approach to hierarchical FRBSs, the variables (and rules) are divided into different levels in such a way that those considered the most influential variables are chosen as input variables at the first level, the next most important variables are chosen as input variables at the second level, and so on. The output variable of each level is introduced as input variable at the following level.

With that structure, the rules at first level of the FRBS have a similar structure to any Mamdani FRBS, i. e., that describe by (13.1), but at k-th level ($k > 1$), rules include the output of the previous level as input

$$\begin{aligned} &\textit{IF } X_{N_k+1} \textit{ is } LT_{N_k+1} \textit{ and } \ldots \textit{ and } X_{N_k+n_k} \textit{ is } LT_{N_k+n_k} \\ &\quad \textit{and } O_{k-1} \textit{ is } LT_{Ok-1} \textit{ THEN } O_k \textit{ is } LT_{Ok} , \end{aligned} \quad (13.28)$$

where the value N_k determines the input variables considered in previous levels

$$N_k = \sum_{t=1}^{k-1} n_t , \quad (13.29)$$

with n_t being the number of system variables applied at level t. Variable O_k represent the output of the k level of the hierarchy. All outputs are intermediate variables except for the output of the last level that will be Y (the overall output of the system).

With this structure it is shown [13.43] that the number of rules in a complete rule base could be reduced to a linear function of the number of variables, while in a conventional FRBS it was an exponential function of the number of variables.

13.4 Fuzzy Rule-Based Systems Design

Once defined the components and functioning of an FRBS, it is time to consider its design, i. e., how to built an FRBS to solve a certain problem while showing some specific properties. The present section will focus on this question.

An FRBS can be characterized according to its structure and its behavior. When referring to its structure, we can consider questions as the dimension of the system (number of variables, fuzzy sets, rules, etc.) as well as other aspects related to properties of its components (distinguishability of the fuzzy sets, redundancy of the fuzzy rules, etc.). On the other hand, the characterization related to the behavior mostly analyzes properties considering the input–output relation defined by the FRBS. In this area, we can include questions as stability or accuracy. Finally, there is a third question that simultaneously involves structure and behavior. This question is interpretability, a central aspect in fuzzy systems design that is considered in an independent chapter.

13.4.1 FRBS Properties

All the structural properties to be mentioned are related to properties of the KB, and basically cover characteristics related to the individual fuzzy sets, the fuzzy partitions related to each input and output variable, the fuzzy rules, and the rule set as a whole.

The elemental components of the KB are fuzzy sets. At this level, we have several questions to be analyzed as normality, convexity, or differentiability of fuzzy sets; all of them being related to the properties of the membership function ($\mu_A(x)$) defining the fuzzy set (A). In most applications the considered fuzzy sets adopt predefined shapes as triangular, trapezoidal, Gaussian, or bell; the fuzzy sets are then defined by only changing some parameters of these *parameterized functions*. In summary, most fuzzy sets considered in FRBSs are normal and convex sets belonging to one of two possible families: piecewise linear functions and differentiable functions. Piecewise linear functions are basically triangular and trapezoidal functions offering a reduced complexity from the processing point of view. On the other hand, differentiable functions are mainly Gaussian, bell, and sigmoidal functions being better adapted to some kind of differential learning approaches as those used in neuro-fuzzy systems, but adding complexity from the processing point of view.

Once individual fuzzy sets have been considered, the following level is that of fuzzy partitions related to each variable. The main characteristics of a fuzzy partition are cardinality, coverage, and distinguishability. Cardinality corresponds to the number of fuzzy sets that compose the fuzzy partitions. In most cases, this number ranges from 3 to 9, with 9 being an upper limit commonly accepted after the ideas of *Miller* [13.44]. The larger the number of fuzzy sets in the partition, the most difficult the design and interpretation of the FRBS. Coverage corresponds to the minimum membership degree with which any value of the variable (x), through its universe of discourse (U), will be assigned to at least a fuzzy set (A_i) in the partition. Coverage is then defined as

$$\min_{x\in U} \max_{i=1\ldots n} \mu_{A_i}(x) , \tag{13.30}$$

being n the cardinality of the partition. As an example, the fuzzy partition in Fig. 13.2 has a coverage of 0.5. Finally, distinguishability of fuzzy sets is related to the level of overlapping of their membership functions, being analyzed with different expressions.

On the basis of the fuzzy sets and fuzzy partitions, the fuzzy rules are built. The first structural question regarding fuzzy rules is the type of fuzzy rule to be considered: Mamdani, TSK, descriptive or approximate, DNF, etc. If we consider now the interaction between the different fuzzy rules of a fuzzy system, questions as knowledge consistency or redundancy appear, i. e., does a fuzzy system include pieces of knowledge (usually rules) providing contradictory (or redundant) information for a specific situation. Finally, when considering the rule base as a whole, completeness and complexity are to be considered. Completeness refers to the fact that any potential input value will fire at least one rule.

Considering now behavioral properties, the most widely analyzed are stability and accuracy. It is also possible to take into account other properties as continuity or robustness, but we will concentrate in those having the larger presence in the literature. Behavioral properties are related to the overall system, i. e., to the processing structure as well as to the KB.

Stability is a key aspect of dynamical systems analysis, and plays a central role in control theory. FRBSs are nonlinear dynamical systems, and after its early application to control problems, the absence of a formal stability analysis was seriously criticized. As a consequence, the stability question received significant attention from the very beginning, at present being a widely studied problem [13.45] for both Mamdani and TSK fuzzy systems, considering the use of different approaches as Lyapunov's methods, Popov criterion or norm-based analysis among others.

Another question with a continuous presence in the literature is that of accuracy and the somehow related concept of universal approximation. The idea of fuzzy systems as universal approximators states that, given any continuous real-valued function on a compact subset of R^n, we can, at least in theory, find an FRBS that approximates this function to any degree. This property has been established for different types of fuzzy systems [13.46–48]. On this basis, the idea of building fuzzy models with an unbounded level of accuracy can be considered. In any case, it is important to notice that previous papers proof the existence of such a model but assuming at the same time an unbounded complexity, i. e., the number of fuzzy sets and rules involved in the fuzzy system will usually grow as the accuracy improves. That means that improving accuracy is possible but always with a cost related either to the complexity of the system or to the relaxation of some of its properties (usually interpretability).

13.4.2 Designing FRBSs

Given a modeling, classification, or control problem to be solved, and assumed it will be focused on through an FRBS, there are several steps in the process of design. The first decision is the choice between the different types of systems mentioned in Sect. 13.2, particularly Mamdani and TSK approaches. They offer different characteristics related to questions as their accuracy and interpretability, as well as different methods for the derivation of its KB.

Once chosen a type of FRBS, its design implies the construction of its processing structure as well as the derivation of its KB. Even considering that there are several options to modify the processing structure of the system (Sect. 13.1.2), most designers consider a standard inference engine and concentrate on the knowledge extraction problem.

Going now to the knowledge extraction problem, some of its parts are common to any modeling process (being fuzzy or not). Questions as the selection of the input and output variables and the determination of the range of those variables are generic to any modeling approach. The specific aspects related to the fuzzy environment are the definition of the fuzzy sets or the fuzzy partition related to each of those variables, and the derivation of a suitable set of fuzzy rules. These two components can be jointly derived in a single process, or sequentially performed by considering first the design of the fuzzy partition associated with each variable and then the fuzzy rules. The design process can be based on two main sources of information: expert knowledge and experimental data.

If we first consider the definition of fuzzy sets and fuzzy partitions, quite different approaches [13.49] can be applied. Even the idea of simply generating a uniformly distributed strong fuzzy partition of a certain cardinality is widely considered.

Going now to rules, Mamdani FRBSs are particularly adapted to expert knowledge extraction, and knowledge elicitation for that kind of system has been widely considered in the literature. In any case, there is not a standard methodology for fuzzy knowledge extraction from experts and at present most practical works consider either a direct data-driven approach, or the integration of expert and data-driven knowledge extraction [13.50].

When considering data-driven knowledge extraction, there is an almost endless list of approaches. Some options are the use of ad-hoc methods based on data covering measures (as [13.46]), the generation of fuzzy decisions trees [13.51], the use of clustering techniques [13.52], and the use of hybrid systems where genetic fuzzy systems [13.53] and neuro fuzzy systems [13.54] represent the most widely considered approaches to fuzzy systems design. Some of those techniques produce both the partitions (or fuzzy sets) and the rules in a single process.

13.5 Conclusions

Fuzzy rule-based systems constitute a tool for representing knowledge and reasoning on it. Jointly with fuzzy clustering techniques, FRBSs are probably the developments of fuzzy sets theory leading to the larger number of applications. These systems, being a kind of rule-based system, can be analyzed as knowledge-based systems showing a structure with two main components: knowledge and processing. The processing structure relays on many concepts presented in previous chapters as fuzzy implications, connectives, relations and so on. In addition, some new concepts as fuzzification and defuzzification are required when constructing a fuzzy rule-based system. But the central concept of fuzzy rule-based systems are fuzzy rules. Different types of fuzzy rules have been considered, particularly those having a fuzzy (or not) consequent, producing different types of FRBS. In addition, new formulations are being considered, e.g., implicative rules. Eventually, the representation capabilities of fuzzy sets have been considered as too limited to represent some specific kinds of knowledge or information, and some extended types of fuzzy sets have been defined. Type-2 fuzzy sets are an example of extension of fuzzy sets.

Having been said that FRBSs are knowledge-based systems, and as a consequence, its design involves, apart from aspects related to the processing structure, the elicitation of a suitable KB properly describing the way to solve the problem under consideration. Even considering the large number of problems solved using FRBSs, there is not a clear design methodology defining a well-established design protocol. In addition, two completely different sources of knowledge, requiring different extraction approaches, have been considered when building FRBSs: expert knowledge and data. Many expert and data-driven knowledge extraction techniques and methods are described in the literature and can be considered. Connected to this question, as part of the process to provide automatic knowledge extraction capabilities to FRBSs, many hybrid approaches have been proposed, genetic fuzzy systems and neuro-fuzzy systems being the most widely considered.

In summary, FRBSs are a powerful tool to solve real world problems, but many theoretical aspects and design questions remain open for further investigation.

References

13.1 A. Bardossy, L. Duckstein: *Fuzzy Rule-Based Modeling with Application to Geophysical, Biological and Engineering Systems* (CRC, Boca Raton 1995)

13.2 Z. Chi, H. Yan, T. Pham: *Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition* (World Scientific, Singapore 1996)

13.3 K. Hirota: *Industrial Applications of Fuzzy Technology* (Springer, Berlin, Heidelberg 1993)

13.4 W. Pedrycz: *Fuzzy Modelling: Paradigms and Practice* (Kluwer Academic, Dordrecht 1996)

13.5 R.R. Yager, L.A. Zadeh: *An Introduction to Fuzzy Logic Applications in Intelligent Systems* (Kluwer Academic, Dordrecht 1992)

13.6 L.A. Zadeh: The concept of a linguistic variable and its applications to approximate reasoning – Part I, Inf. Sci. **8**(3), 199–249 (1975)

13.7 L.A. Zadeh: The concept of a linguistic variable and its applications to approximate reasoning – Part II, Inf. Sci. **8**(4), 301–357 (1975)

13.8 L.A. Zadeh: The concept of a linguistic variable and its applications to approximate reasoning – Part III, Inf. Sci. **9**(1), 43–80 (1975)

13.9 L.A. Zadeh: Outline of a new approach to the analysis of complex systems and decision processes, IEEE Trans. Syst. Man Cybern. **3**, 28–44 (1973)

13.10 E.H. Mamdani: Applications of fuzzy algorithm for control of simple dynamic plant, Proc. IEE **121**(12), 1585–1588 (1974)

13.11 E.H. Mamdani, S. Assilian: An experiment in linguistic synthesis with a fuzzy logic controller, Int. J. Man-Mach. Stud. **7**, 1–13 (1975)

13.12 L.X. Wang: *Adaptive Fuzzy Systems and Control: Design and Analysis* (Prentice Hall, Englewood Cliffs 1994)

13.13 T.J. Procyk, E.H. Mamdani: A linguistic self-organizing process controller, Automatica **15**(1), 15–30 (1979)

13.14 L. Magdalena: Adapting the gain of an FLC with genetic algorithms, Int. J. Approx. Reas. **17**(4), 327–349 (1997)

13.15 G.C. Mouzouris, J.M. Mendel: Nonsingleton fuzzy logic systems: Theory and application, IEEE Trans. Fuzzy Syst. **5**, 56–71 (1997)

13.16 F. Klawonn, R. Kruse: Equality relations as a basis for fuzzy control, Fuzzy Sets Syst. **54**(2), 147–156 (1993)

13.17 J.M. Mendel: Fuzzy logic systems for engineering: A tutorial, Proc. IEEE **83**(3), 345–377 (1995)

13.18 D. Dubois, H. Prade: What are fuzzy rules and how to use them, Fuzzy Sets Syst. **84**, 169–185 (1996)

13.19 O. Cordón, F. Herrera, A. Peregrín: Applicability of the fuzzy operators in the design of fuzzy logic controllers, Fuzzy Sets Syst. **86**, 15–41 (1997)

13.20 D. Driankov, H. Hellendoorn, M. Reinfrank: *An Introduction to Fuzzy Control* (Springer, Berlin, Heidelberg 1993)

13.21 M. Sugeno, T. Yasukawa: A fuzzy-logic-based approach to qualitative modeling, IEEE Trans. Fuzzy Syst. **1**(1), 7–31 (1993)

13.22 C.C. Lee: Fuzzy logic in control systems: Fuzzy logic controller – Part I, IEEE Trans. Syst. Man Cybern. **20**(2), 404–418 (1990)

13.23 C.C. Lee: Fuzzy logic in control systems: Fuzzy logic controller – Part II, IEEE Trans. Syst. Man Cybern. **20**(2), 419–435 (1990)

13.24 A. Bastian: How to handle the flexibility of linguistic variables with applications, Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **3**(4), 463–484 (1994)

13.25 B. Carse, T.C. Fogarty, A. Munro: Evolving fuzzy rule based controllers using genetic algorithms, Fuzzy Sets Syst. **80**, 273–294 (1996)

13.26 A. González, R. Pèrez, J.L. Verdegay: Learning the structure of a fuzzy rule: A genetic approach, Fuzzy Syst. Artif. Intell. **3**(1), 57–70 (1994)

13.27 L. Magdalena, F. Monasterio: A fuzzy logic controller with learning through the evolution of its knowledge base, Int. J. Approx. Reas. **16**(3/4), 335–358 (1997)

13.28 R. Alcalá, J. Casillas, O. Cordón, F. Herrera: Building fuzzy graphs: Features and taxonomy of learning for non-grid-oriented fuzzy rule-based systems, J. Intell. Fuzzy Syst. **11**(3/4), 99–119 (2001)

13.29 O. Cordón, F. Herrera: A three-stage evolutionary process for learning descriptive and approximate fuzzy logic controller knowledge bases from examples, Int. J. Approx. Reas. **17**(4), 369–407 (1997)

13.30 L. Koczy: Fuzzy if ... then rule models and their transformation into one another, IEEE Trans. Syst. Man Cybern. **26**(5), 621–637 (1996)

13.31 T. Takagi, M. Sugeno: Fuzzy identification of systems and its application to modeling and control, IEEE Trans. Syst. Man Cybern. **15**(1), 116–132 (1985)

13.32 M. Sugeno, G.T. Kang: Structure identification of fuzzy model, Fuzzy Sets Syst. **28**(1), 15–33 (1988)

13.33 R. Palm, D. Driankov, H. Hellendoorn: *Model Based Fuzzy Control* (Springer, Berlin, Heidelberg 1997)

13.34 H. Ishibuchi, T. Nakashima: Effect of rule weights in fuzzy rule-based classification systems, IEEE Trans. Fuzzy Syst. **9**, 506–515 (2001)

13.35 J.M. Mendel: Type-2 fuzzy sets and systems: An overview, Comput. Intell. Mag. IEEE **2**(1), 20–29 (2007)

13.36 H. Jones, B. Charnomordic, D. Dubois, S. Guillaume: Practical inference with systems of gradual implicative rules, IEEE Trans. Fuzzy Syst. **17**, 61–78 (2009)

13.37 V. Torra: A review of the construction of hierarchical fuzzy systems, Int. J. Intell. Syst. **17**(5), 531–543 (2002)

13.38 R.R. Yager: On a hierarchical structure for fuzzy modeling and control, IEEE Trans. Syst. Man Cybern. **23**(4), 1189–1197 (1993)

13.39 R.R. Yager: On the construction of hierarchical fuzzy systems models, IEEE Trans. Syst. Man Cybern. C **28**(1), 55–66 (1998)

13.40 O. Cordón, F. Herrera, I. Zwir: Linguistic modeling by hierarchical systems of linguistic rules, IEEE Trans. Fuzzy Syst. **10**, 2–20 (2002)

13.41 E. D'Andrea, B. Lazzerini: A hierarchical approach to multi-class fuzzy classifiers, Exp. Syst. Appl. **40**(9), 3828–3840 (2013)

13.42 H. Takagi, N. Suzuki, T. Koda, Y. Kojima: Neural networks designed on approximate reasoning architecture and their applications, IEEE Trans. Neural Netw. **3**(5), 752–760 (1992)

13.43 G.V.S. Raju, J. Zhou, R.A. Kisner: Hierarchical fuzzy control, Int. J. Control **54**(5), 1201–1216 (1991)

13.44 G.A. Miller: The magical number seven, plus or minus two: Some limits on our capacity for processing information, Psychol. Rev. **63**, 81–97 (1956)

13.45 K. Michels, F. Klawonn, R. Kruse, A. Nürnberger: *Fuzzy Control: Fundamentals, Stability and Design of Fuzzy Controllers* (Springer, Berlin, Heidelberg 2006)

13.46 L.-X. Wang, J.M. Mendel: Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, IEEE Trans. Neural Netw. **3**(5), 807–813 (1992)

13.47 B. Kosko: Fuzzy systems as universal approximators, IEEE Trans. Comput. **43**(11), 1329–1333 (1994)

13.48 J.L. Castro: Fuzzy logic controllers are universal approximators, IEEE Trans. Syst. Man Cybern. **25**(4), 629–635 (1995)

13.49 R. Krishnapuram: Membership function elicitation and learning. In: *Handbook of Fuzzy Computation*, ed. by E.H. Ruspini, P.P. Bonissone, W. Pedrycz (IOP Publ., Bristol 1998) pp. 349–368

13.50 J.M. Alonso, L. Magdalena: HILK++: An interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers, Soft Comput. **15**(10), 1959–1980 (2011)

13.51 N.R. Pal, S. Chakraborty: Fuzzy rule extraction from id3-type decision trees for real data, IEEE Trans. Syst. Man Cybern. B **31**(5), 745–754 (2001)

13.52 M. Delgado, A.F. Gómez-Skarmeta, F. Martín: A fuzzy clustering-based rapid prototyping for fuzzy rule-based modeling, IEEE Trans. Fuzzy Syst. **5**, 223–233 (1997)

13.53 O. Cordón, F. Herrera, F. Hoffmann, L. Magdalena: *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (World Scientific, Singapore 2001)

13.54 D.D. Nauck, A. Nürnberger: Neuro-fuzzy systems: A short historical review. In: *Computational Intelligence in Intelligent Data Analysis*, ed. by C. Moewes, A. Nürnberger (Springer, Berlin, Heidelberg 2013) pp. 91–109